

IN THE CLAIMS

1. (Previously Presented) A method for providing redundant data load sharing in a distributed network, comprising:

receiving an original entry;

storing the original data entry in a first one of a plurality of nodes;

generating a replicated data entry from the original data entry;

storing the replicated data entry in a second one of the plurality of nodes;

identifying a failure of either of the first one or the second one of the plurality of nodes;

generating a re-replicated data entry in the non-failed first or second one of the plurality of nodes for storage at a third one of the plurality of nodes in response to the failure in the first or second one of the plurality of nodes.

2. (Previously Presented) The method of Claim 1, further comprising:

determining whether there is sufficient capacity in the distributed network to handle data entry replication in response to the failure of the first or second one of the plurality of nodes.

3. (Previously Presented) The method of Claim 2, further comprising:

preventing replication of the original or replicated data entry at the third one of the plurality of nodes in response to insufficient capacity in the distributed network.

4. (Previously Presented) The method of Claim 3, further comprising:

dynamically adjusting the capacity of the distributed network in response to the failure of the first or second one of the plurality of nodes in order to store new data entries without replication.

5. (Previously Presented) The method of Claim 3, further comprising:

identifying a recovery of the failed first or second one of the plurality of nodes;

performing storage and replication of subsequently received data entries in response to the recovery of the failed first or second one of the plurality of nodes.

6. (Previously Presented) The method of Claim 5, further comprising:

adjusting the capacity of the distributed network in response to the recovery of the failed first or second one of the plurality of nodes.

7. (Previously Presented) The method of Claim 5, further comprising:

performing replication of those data entries previously stored but not replicated as a result of the failure of the first or second one of the plurality of nodes.

8. (Previously Presented) The method of Claim 1, further comprising:

identifying a recovery of the failed first or second one of the plurality of nodes;

including the recovered failed first or second one of the plurality of nodes in the storage and replication of subsequent data entries.

9. (Original) The method of Claim 1, further comprising:

establishing a capacity for the distributed network, the capacity representing an amount of data to be stored in the distributed network;

establishing a minimum number of the plurality of nodes required to provide redundancy in the distributed network.

10. (Original) The method of Claim 9, wherein at least a single occurrence of all data entries are maintained in the plurality of nodes when the number of plurality of nodes falls to one less than the minimum number.

11. (Previously Presented) A system for providing redundant data load sharing in a distributed network, comprising:

a plurality of nodes, a first one of the plurality of nodes operable to receive and store an original data entry, the first one of the plurality of nodes operable to provide a replicate data entry to a second one of the plurality of nodes, the first one of the plurality of nodes operable to determine a failure of the second one of the plurality of nodes, the first one of the plurality of nodes operable to provide a re-replicated data entry to a third one of the plurality of nodes in response to failure of the second one of the plurality of nodes.

12. (Previously Presented) The system of Claim 11, wherein each node includes a distributed control function operable to control storage and replication of the original data entry.

13. (Original) The system of Claim 11, wherein the first one of the plurality of nodes is operable to determine whether there is sufficient capacity in the distributed network to handle data entry replication in response to the failure of the second one of the plurality of nodes.

14. (Previously Presented) The system of Claim 13, wherein the first one of the plurality of nodes is operable to prevent replication of the original data entry at the third one of the plurality of nodes in response to insufficient capacity in the distributed network.

15. (Original) The system of Claim 14, wherein the plurality of nodes are operable to dynamically adjust the capacity of the distributed network in response to the failure of the second one of the plurality of nodes in order to store new data entries without replication.

16. (Previously Presented) The system of Claim 11, wherein the capacity of each of the plurality of nodes is adjusted in response to an addition of a new node or failure of an existing node.

17. (Previously Presented) A system for providing redundant data load sharing in a distributed network, comprising:

means for receiving an original entry;

means for storing the original data entry in a first one of a plurality of nodes;

means for generating a replicated data entry from the original data entry;

means for storing the replicated data entry in a second one of the plurality of nodes;

means for identifying a failure of either of the first one or the second one of the plurality of nodes;

means for generating a re-replicated data entry in the non-failed first or second one of the plurality of nodes for storage at a third one of the plurality of nodes in response to the failure in the first or second one of the plurality of nodes.

18. (Previously Presented) The system of Claim 17, further comprising:

means for determining whether there is sufficient capacity in the distributed network to handle data entry replication in response to the failure of the first or second one of the plurality of nodes.

19. (Previously Presented) The system of Claim 18, further comprising:

means for preventing replication of the original or replicated data entry at the third one of the plurality of nodes in response to insufficient capacity in the distributed network.

20. (Previously Presented) The system of Claim 19, further comprising:

means for dynamically adjusting the capacity of the distributed network in response to the failure of the first or second one of the plurality of nodes in order to store new data entries without replication.

21. (Previously Presented) The system of Claim 19, further comprising:

means for identifying a recovery of the failed first or second one of the plurality of nodes;

means for performing storage and replication of subsequently received data entries in response to the recovery of the failed first or second one of the plurality of nodes.

22. (Previously Presented) A computer readable medium including code for providing redundant data load sharing in a distributed network, the code operable to:

receive an original entry;

store the original data entry in a first one of a plurality of nodes;

generate a replicated data entry from the original data entry;

store the replicated data entry in a second one of the plurality of nodes, the original data entry and the replicated data entry including information as to where it is redundantly stored;

identify a failure of either of the first one or the second one of the plurality of nodes;

generating a re-replicated data entry in the non-failed first or second one of the plurality of nodes for storage at a third one of the plurality of nodes in response to the failure in the first or second one of the plurality of nodes.

23. (Previously Presented) The computer readable medium of Claim 22, wherein the code is further operable to:

determine whether there is sufficient capacity in the distributed network to handle data entry replication in response to the failure of the first or second one of the plurality of nodes.

24. (Previously Presented) The computer readable medium of Claim 23, wherein the code is further operable to:

prevent replication of the original or replicated data entry at the third one of the plurality of nodes in response to insufficient capacity in the distributed network.

25. (Previously Presented) The computer readable medium of Claim 24, wherein the code is further operable to:

dynamically adjust the capacity of the distributed network in response to the failure of the first or second one of the plurality of nodes in order to store new data entries without replication.

26. (Previously Presented) The computer readable medium of Claim 24, wherein the code is further operable to:

identify a recovery of the failed first or second one of the plurality of nodes;

perform storage and replication of subsequently received data entries in response to the recovery of the failed first or second one of the plurality of nodes.